LEGIBILITY NOTICE

A major purpose of the Technical Information Center is to provide the broadest dissemination possible of information contained in DOE's Research and Development Reports to business, industry, the academic community, and federal, state and local governments.

Although a small portion of this report is not reproducible, it is being made available to expedite the availability of information on the research discussed herein.

LA-UR-87-87

LA-UR--87-87

DE87 005112

5

Los Alamos National Laboratory is operated by the University of California for the United States Department of Energy under contract W-7405-ENG-36

TITLE THE S_N/MONTE CARLO RESPONSE MATRIX HYBRID METHOD

AUTHOR(S): W. L. Filippone, Department of Nuclear and Energy Engineering
University of Arizona, Tucson, AZ 85721

R. E. Alcouffe, X-6, MS B226, Los Alamos National Laboratory
Los Alamos, NM 87545

SUBMITTED TO To be included in the proceedings, Math and Computations Division of the American Nuclear Society conference, Paris, FRANCE, April 27-30, 1987.

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

By acceptance of this article, the publisher recognizes that the U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or to allow others to do so, for U.S. Government purposes.

The Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy

MASTER

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

Los Alamos National Laboratory
Los Alamos, New Mexico 87545

FORM NO. 836 R4
ST. NO. 2629 5/81

THE S_N/MONTE CARLO RESPONSE MATRIX HYBRID METHOD

W. L. Filippone
Department of Nuclear and Energy Engineering
University of Arizona
Tucson, AZ 85721
(602) 621-6165

R. E. Alcouffe
P. O. Box 1663
Group X-6, MS B226
Los Alamos National Laboratory
Los Alamos, NM 87545
(505) 667-5956

ABSTRACT

A hybrid method has been developed to iteratively couple S_N and Monte Carlo regions of the same problem. This technique avoids many of the restrictions and limitations of previous attempts to do the coupling and results in a general and relatively efficient method. We demonstrate the method with some simple examples.

I. INTRODUCTION

The S_N/Monte Carlo hybrid method consists of defining spatial regions of a problem as regions where either a Monte Carlo calculation or an S_N calculation is to be done. The calculations are then connected through the common boundary fluxes. Operationally, this connection or linkage is accomplished through the use of response matrices in the Monte Carlo regions. These response matrices give the angular flux of particles leaving the Monte Carlo regions due to the angular flux of particles entering the Monte Carlo regions. The response matrices are calculated using Monte Carlo and are stored to be used in a boundary flux iteration. The boundary sources into the Monte Carlo regions are computed in the S_N regions and, thus, are accurate only when the solution on the boundary out of the Monte Carlo regions is know. The boundary fluxes must, therefore, be iterated between the Monte Carlo and S_N regions.

To facilitate the linkage between the S_N and Monte Carlo parts of the calculation, all interface fluxes are defined on the S_N numerical grid. Therefore, the response operator for the i'th Monte Carlo region is a $K_i \times K_i$ matrix, where K_i is the number (spatial times angular) of S_N states used to represent the boundary flux for the region.

At each iteration step the exiting fluxes, due to the current estimate of the incident fluxes, are to be determined. Precalculated response matrices are used only in the Monte Carlo regions. In the S_N regions, the exiting fluxes are determined by numerically solving the transport equation.

Precalculated response matrices are not used in the S_N regions since they would increase both storage and computational requirements. Computing a K×K response matrix is K times more costly than a single boundary value problem. Since typical values of K are much larger than the required number of interface flux iterations, precalculating response matrices for the S_N regions is not advisable.

The situation is quite different for the Monte Carlo regions. By recording the initial (S_N) state in addition to the final (S_N) state of each particle history, a single Monte Carlo calculation can generate an entire response matrix instead of just an exit flux. As explained in Ref. 1, such a calculation requires roughly the same computation time as a single boundary value problem of comparable accuracy. (The accuracy of concern here is not that of the individual response matrix elements, which will be relatively poor, but rather that of the exit fluxes calculated directly and via the response matrix.)

The hybrid method has been implemented in a computer code called TWODANT/MC. This was done by adding Monte Carlo and response matrix linkage modules to TWODANT.[2] Although the technique should be applicable to arbitrary geometries, the present test version of TWODANT/MC is restricted to media consisting of three rectangular regions (see Fig. 1). The top and bottom regions (which need not be homogeneous) are analyzed using S_N, while the center region, including the boundary layer zones, is treated with the Monte Carlo method. The interface fluxes at the top and bottom of the boundary layers are determined iteratively using the response matrix method.

Several test problems have been run with excellent results. The principal limitation of the method appears to be the computer memory available for storing the Monte Carlo response matrices.

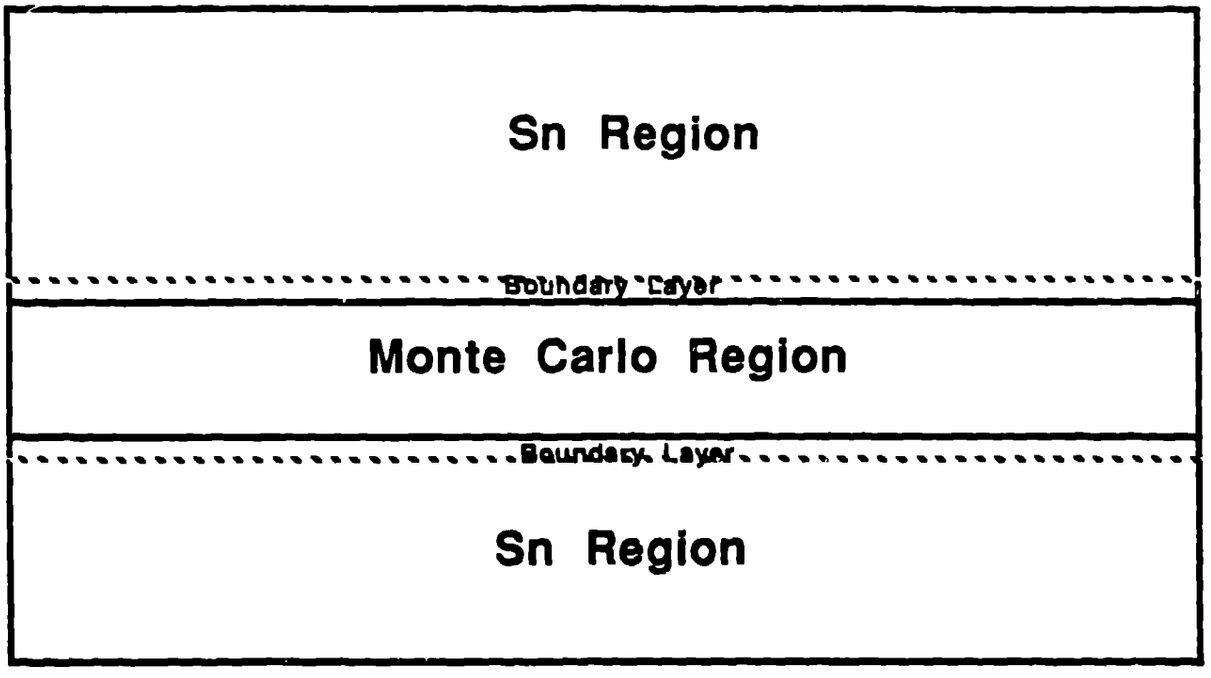

Fig. I. Geometry for the Testing of the Sn-Monte Carlo Link.

II. THEORY

Consider the medium depicted in Fig. I, consisting of one Monte Carlo and two S_N regions. Suppose there are IL S_N spatial mesh cells along each S_N/Monte Carlo interface. Then, the interface fluxes will be K-dimensional vectors where

$$K = IL \times \frac{M}{2} \quad , \tag{1}$$

and

$$M = \text{the total number of angular directions in a two-dimensional } S_N \text{ calculation.} \tag{2}$$

Let the fluxes at the top interface be denoted by $\vec{\psi}_1^{out}$ and $\vec{\psi}_1^{in}$ and those at the bottom as $\vec{\psi}_2^{out}$ and $\vec{\psi}_2^{in}$ where the superscripts refer to directions in or out of the Monte Carlo region. The transmission matrices T_1 and T_2 and the reflection matrices R_1 and R_2 for the Monte Carlo region are defined such that

$$\vec{\psi}_1^{out} = R_1\vec{\psi}_1^{in} + T_2\vec{\psi}_2^{in} + \vec{S}_1^{out} \quad , \tag{3}$$

and

$$\vec{\psi}_2^{out} = R_2\vec{\psi}_2^{in} + T_1\vec{\psi}_1^{in} + \vec{S}_2^{out} \quad , \tag{4}$$

where $\vec{S}_1^{out}$ and $\vec{S}_2^{out}$ are the flux of particles that would leave the top and bottom surfaces of the Monte Carlo region under vacuum ($\vec{\psi}_1^{in} = \vec{\psi}_2^{in} = 0$) boundary conditions. These fluxes result from sources located inside the Monte Carlo region.

Equations (3) and (4) can be solved iteratively:

$$\vec{\psi}_1^{out(P+1)} = R_1 \vec{\psi}_1^{in(P)} + T_2 \vec{\psi}_2^{in(P)} + \vec{S}_1^{out} , \tag{5}$$

$$\vec{\psi}_2^{out(P+1)} = R_2 \vec{\psi}_2^{in(P)} + T_1 \vec{\psi}_1^{in(P)} + \vec{S}_2^{out} . \tag{6}$$

The matrices T_1, T_2, R_1, and R_2 and fluxes $\vec{S}_1^{out}$ and $\vec{S}_2^{out}$ are calculated once using Monte Carlo and then saved for future use. The fluxes $\vec{\psi}_1^{in(P)}$ and $\vec{\psi}_2^{in(P)}$ are calculated at each iteration step using S_N with the prescribed boundary fluxes $\vec{\psi}_1^{out(P)}$ and $\vec{\psi}_2^{out(P)}$.

A. Choosing the S_N and Monte Carlo Regions

Three criteria are important when deciding where to locate the S_N/Monte Carlo interfaces:

1. The S_N regions should be comprised of geometrically simple and relatively highly scattering materials so that the S_N/DSA solver will be fast and accurate.

2. The Monte Carlo region should be as optically thin as possible so that the Monte Carlo calculations are reasonably fast. It may also contain localized sources in phase space or geometrically complex objects.

3. The Monte Carlo/S_N interfaces should be located where the flux is a slowly varying function of angle and position. This is needed to minimize the number of S_N states needed to represent the interface fluxes, thereby minimizing the size of the response matrices.

B. The Boundary Layers

If it were not for Item 3, above, the ideal locations for the S_N/Monte Carlo interfaces would be at the physical boundaries between high- and low-scattering regions. However, this could lead to very large response matrices since the flux near a low-scattering region can be quite anisotropic. For this reason, we have found it advisable to locate the S_N/Monte Carlo interfaces in highly scattering regions about 1 mfp away from a region of low scattering. Thus, for best results, the Monte Carlo region should be comprised of a zone of low-scattering material sandwiched between boundary layers (about 1-mfp thick) with large scattering cross sections.

One obvious drawback in using the boundary layers is that they increase the optical thickness of the Monte Carlo region and, therefore, increase the average number of collisions per history. However, it is our opinion that the reduction in

the amount of storage needed for the response matrices more than compensates for the increase in Monte Carlo computation time.

C. Implementation into TWODANT

The implementation of this method into the TWODANT code involves providing the logic for the interface flux iteration as characterized by Eqs. (5) and (6), while including subroutines to perform the Monte Carlo computation and to compute the response matrices on the S_N spatial and angular mesh. From the S_N point of view, the only difference between the hybrid method and the normal problem-solving process is the inclusion of interior boundary sources. That is, the procedure we use is to set up the entire problem as an S_N problem with the addition of interior boundary conditions. In this way, the logic for the solution of the transport flux remains the same (and, hence, the coding doesn't change), and the DSA accelerator[3] is also unchanged. We, thus, retain the power and computational efficiency of the S_N solver while we provide a linking with the Monte Carlo region. This is an important point and greatly enhances the practicality of this hybrid method.

To illustrate the implementation of the interior boundary conditions, we refer to Fig. II, which depicts a typical S_N spatial mesh. We indicate a direction $\underline{\Omega}_m$, which defines a sweeping direction (down and in, in the case shown). The sweeping starts at the top and right boundaries and continues until Surfaces 3 and 4 are encountered; the S_N values for these surfaces are then replaced by the ψ_m^{out} given by the Monte Carlo calculation, and the normal sweeping resumes. A similar procedure

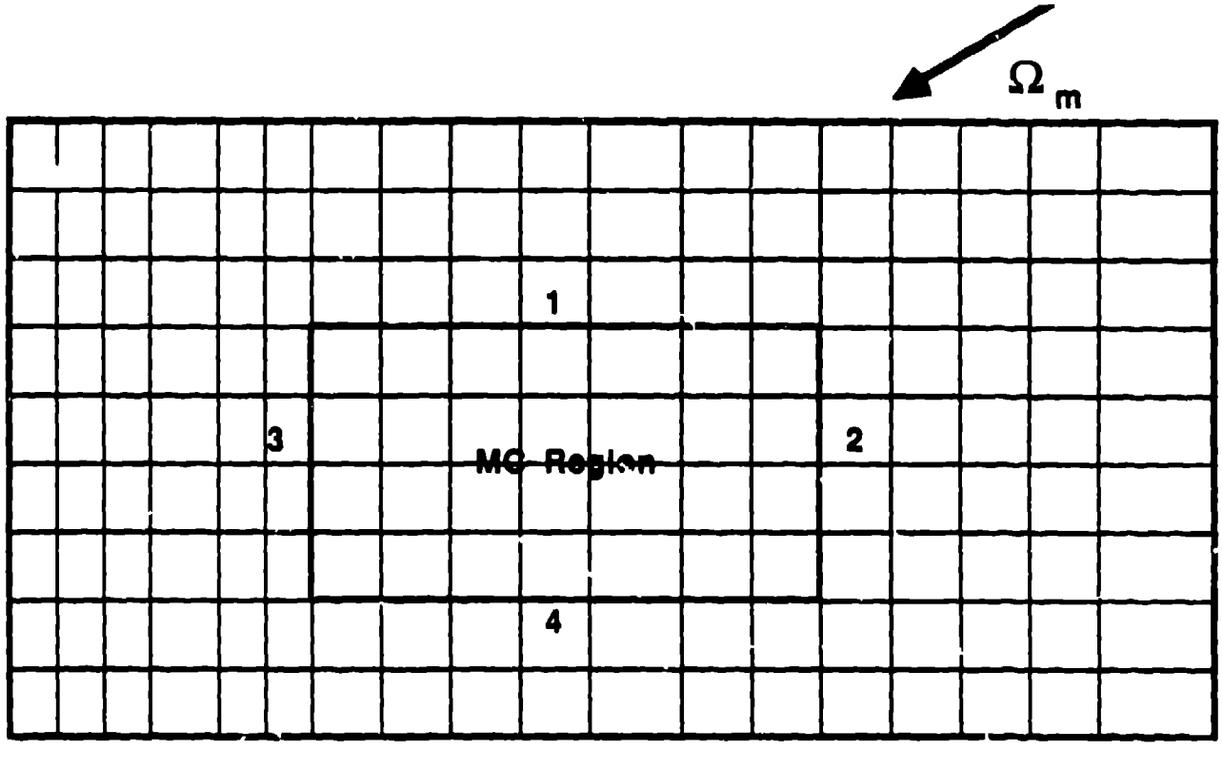

Fig. II. Typical Sn Mesh with Interior Boundaries for MC Coupling.

is followed for the other three directions. This logic is readily implemented, and vectorization of the sweeping algorithm is retained. Thus, the speed of the normal S_N computation is retained. In the same way, the diffusion acceleration equations retain their normal structure; the only difference is the inclusion of the interior boundary sources.

D. Accelerating the Interface Flux Iterations

Although DSA accelerates the S_N calculations, it is not applicable to Eqs. (5) and (6). To reduce the number of interface flux iterations, we use an extrapolation procedure,* which we refer to as the method of residual expansion functions. To describe this technique, it will be helpful to express the interface flux iterations by a single equation. Formally, we can express the S_N calculations as

$$\vec{\psi}_1^{in(P)} = R_1^{SN}\, \vec{\psi}_1^{out(P)} \quad , \tag{7}$$

and

$$\vec{\psi}_2^{in(P)} = R_2^{SN}\, \vec{\psi}_2^{out(P)} \quad , \tag{8}$$

although the reflection matrices R_1^{SN} and R_2^{SN} for the top and bottom S_N regions are not actually determined.

Using Eqs. (7) and (8), Eqs. (5) and (6) can be expressed by the single equation

$$\vec{\psi}^{out(P+1)} = B\vec{\psi}^{out(P)} + \vec{S}^{out} \quad , \tag{9}$$

where

$$B = \begin{bmatrix} q_1 R_1^{SN} & T_2 R_2^{SN} \\ T_1 R_1^{SN} & R_2 R_2^{SN} \end{bmatrix} \quad , \tag{10}$$

$$\vec{\psi}^{out} = \begin{bmatrix} \vec{\psi}_1^{out} \\ \vec{\psi}_2^{out} \end{bmatrix} \quad , \tag{11}$$

$$\vec{S}^{out} = \begin{bmatrix} \vec{S}_1^{out} \\ \vec{S}_2^{out} \end{bmatrix} \quad , \tag{12}$$

and P is the iteration index. The residual,

$$\vec{r}^{(P)} = \vec{\psi}^{out(P+1)} - \vec{\psi}^{out(P)} \quad . \tag{13}$$

It follows from Eqs. (9) and (13) that

$$\vec{r}^{(P+1)} = B\vec{r}^{(P)} = B^P\vec{r}^{(1)} = \sum_{i=1}^{q} a_i \lambda_i^P \vec{y}_i \quad , \tag{14}$$

where

λ_i = the eigenvalue of B with the i'th largest magnitude,

$\vec{y}_i$ = the eigenvector corresponding to λ_i,

q = the number of values of i for which $a_i\lambda_i^P$ is not negligible compared to $a_1\lambda_1^P$, and

a_i = the i'th coefficient in the expansion of $\vec{r}^{(1)}$ in terms of the $\vec{y}_i$.

Here we are assuming for simplicity that B has a complete set of eigenvectors. We Refer to the condition defined by approximation (14) as q'th order shape convergence at iteration p.

It can be shown that

$$\vec{\psi}^{out} = \vec{\psi}^{out(P+q+1)} - \sum_{i=1}^{q} \alpha_i \vec{r}^{(P+i+1)} \quad , \tag{15}$$

where

$$\alpha_i = \frac{\sum_{\ell=1}^{i} \beta_\ell}{\sum_{\ell=1}^{q} \beta_\ell - 1} \quad , \tag{16}$$

and the β_ℓ are obtained from a least squares fit to the equation

$$\vec{r}^{(p+q+1)} = \sum_{\ell=1}^{q} \beta_\ell \vec{r}^{(P+\ell)} \quad . \tag{17}$$

Exact equality holds in Eqs. (15) and (17) whenever exact equality holds in Eq. (14).

To predict the converged solution via Eq. (15) requires a total of

$$\underline{P} = p + q + 1 \tag{18}$$

iterations. For any realistic problem, the relationship between p and q is not known apriori. Therefore, to implement the method, we fix a value for P (typical choices are from 5 to 8) and then test different values of q in Eq. (17), beginning with q = 1. The accuracy of the fit should improve until the correct value of q is exceeded, at which time the residuals on the right-hand side of Eq. (17) become linearly dependent and the least squares problem becomes singular. Typical values of q range from 1 to 5.

III. EXAMPLE CALCULATIONS

In the following, we present two example calculations on systems designed to test the accuracy and calculational efficiency of this method. What we seek to test is sensitivity to the S_N quadrature used and the number of boundary iterations required for an accurate solution. It is clear that the number of boundary flux iterations depends upon the scattering ratio in the S_N region; if c = 0, then no iterations are required; and for c = 1, we expect the number of iterations to be maximum. Thus, referring to Fig. I, we set the scattering ratio for the S_N regions to be unity (c = 1). In our test problems, we then vary the Monte Carlo region size and the source.

For Problem 1 (referring to Fig. I), we prescribe the system as follows:

1. Geometry (X-Y), 10 cm × 25 cm.
   The Y dimension of each S_N region is 10 cm, and the Y dimension of the Monte Carlo region is 5 cm.

2. Material (homogeneous).
   The total cross sections in the S_N regions is 1 cm^{-1} and is 0.1 cm^{-1} in the Monte Carlo region; c = 0.1 in the Monte Carlo region.

3. The source is an isotropic point source normalized to one particle at position (X = 1.0, Y = 12.5), which is in the middle of the Monte Carlo region near the left boundary.

4. Spatial Mesh, 20 × 50 uniform spacing.

The Monte Carlo calculation starts 200,000 particles from the source and about 200,000 from the region boundaries to compute the source contribution to the boundary flux and the response matrices, respectively. In the S_N calculation, we varied the S_N order from N = 2 to N = 8. For the results, we present the leakage current from each spatial cell at the right-hand boundary of the system. In order to judge the accuracy of the results, we also performed benchmarking calculations using MCNP[5] to obtain the Monte Carlo solution for the whole problem and also S-50 for an S_N benchmark.

In Fig. III we present the results of three calculations; the benchmark MCNP, the benchmark S-50, and the hybrid S_N/MC. We see quite good agreement in all three calculations for the leakage current; the greatest discrepancy occurring in the "slot" or Monte Carlo region. In Fig. IV we expand this region and note that the MCNP and hybrid calculations agree very well while the S-50 is flatter across the slot. This flatness arises because we must represent a point source by a volume source in a mesh cell; thus, the discrepancy is caused by the finite extent of the source. We also note that the solution should be symmetric about Y = 12.5, but the

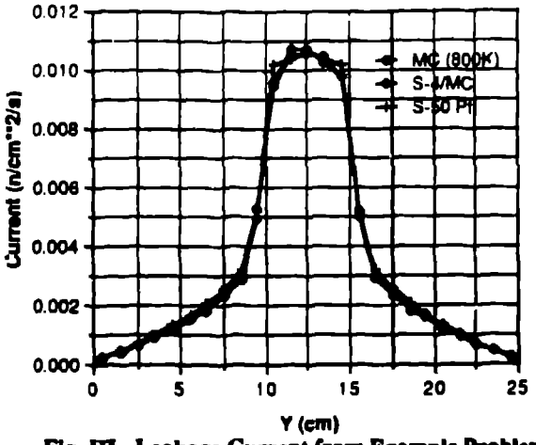

Fig. III. Leakage Current from Example Problem 1.

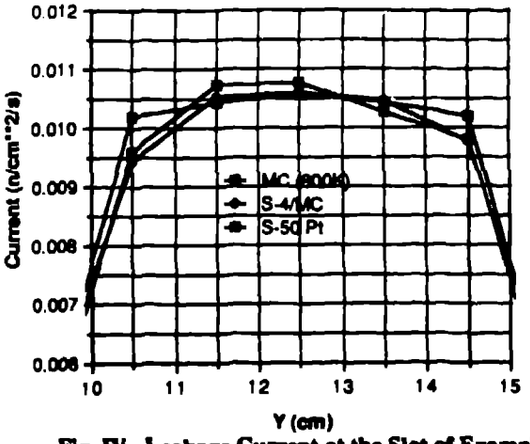

Fig. IV. Leakage Current at the Slot of Example Problem 1.

MCNP and hybrid calculations are a bit skewed, which we assume is statistical error. All-in-all, the agreement is quite good with a maximum deviation of -3%. To contrast with this, in Fig. V, we present calculations varying the S_N order without the Monte Carlo hybrid and see that the errors in the slot are substantial even up to S-30. Next, in Fig. VI, we show the leakage current from a series of S_N/Monte Carlo hybrid calculations where we have varied the S_N order from S-2 to S-8. It is seen that the results are not very sensitive to S_N order in this problem. Finally, we present results that show the convergence of the solution in the slot area as a

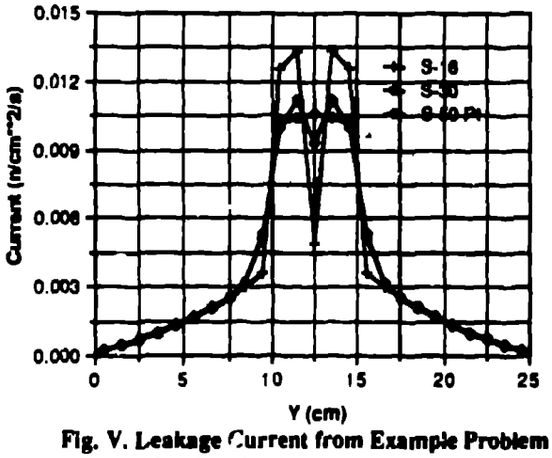

Fig. V. Leakage Current from Example Problem 1 Varying the Sn Order.

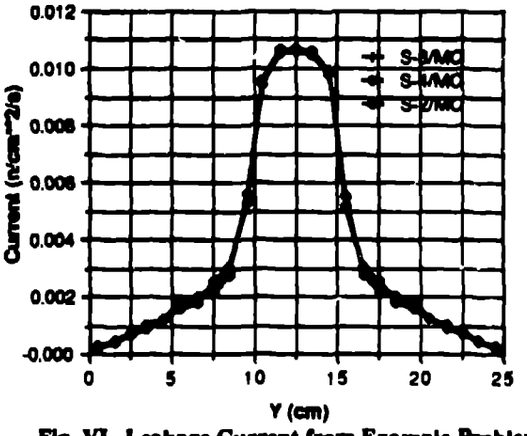

Fig. VI. Leakage Current from Example Problem 1 Varying the Hybrid Sn Order.

function of boundary flux iteration [Eqs. (1) and (2)] in Fig. VII. We note that, in obtaining these results, we have used the projection method described in Section II.D. For this example, it is seen that the answer is converged by iteration 2 ($\underline{P} = 2$), which is good news indeed. To ascertain the effect of our projecting the solution at the end of the iteration cycle, we present iteration results in Fig. VIII in which the solution is not projected. In this case, we require five iterations to converge, which is a factor of 2.5 greater.

For our next example, we provide a more stringent test of the method because of a very tight coupling between the regions. We prescribe this system as follows (again, refer to Fig. I):

1. Geometry (X-Y), 20 cm × 21 cm.
   The Y dimension of each S_N region is 10 cm and the Y dimension of the Monte Carlo region is 1 cm.

2. Material (homogeneous).
   Same as example in Problem 1.

3. Source.
   Isotropic boundary source, normalized to one particle located at ($X = 0.0, 10.0 = Y = 11.0$), that is, located at the left-hand boundary of the MC region.

4. Spatial mesh.
   20 × 42 uniform spacing.

Again, the Monte Carlo calculation starts 200,000 particles. Thus, in this problem, we have a long, narrow slot surrounded by perfect scattering regions. To obtain benchmark solutions, we performed an MCNP calculation and an S-100 calculation of the system. In Fig. IX we present the MCNP calculation, an S-100 calculation, and the hybrid S-4/Monte Carlo calculation of the leakage current. Again, as in the first example, we see very good agreement among the solutions even for this more difficult problem. The value of the current at the slot is or particular interest; this is the point at 10.5 cm on the plot. To see the sensitivity to a normal S_N calculation, we plot the results as a function of S_N order in Fig. X. The sensitivity at the slot is quite strong, which is well known in normal S_N calculations for problems of this type. to ascertain the sensitivity of the hybrid method, we show the same results from S_2 to S_{12} in Fig. XI. There is more sensitivity here than in Example 1; but the results, even with S_2, are quite acceptable. The maximum difference in the slot current is 3.4% from S-12/Monte Carlo to S-4/Monte Carlo, while there is an enormous savings in storage when using the very acceptable S-4/Monte Carlo. As we did for Example 1, we watch the convergence of the solution to Example 2 in the hybrid S_N/Monte Carlo mode in the neighborhood of the slot; the results are shown in Fig. XII. Here, we notice that the exit currents converge more slowly; but by five iterations (P=5), we are well converged for our purposes. Again, to see the effect of the projection of the solution, we display in Fig. XIII the values of the current as a function of iteration without the projection. Comparing Figs. XII and XIII leads us to conclude that the projection saves between a factor of 2 to 3 in the number of iterations required and, hence, is very worthwhile.

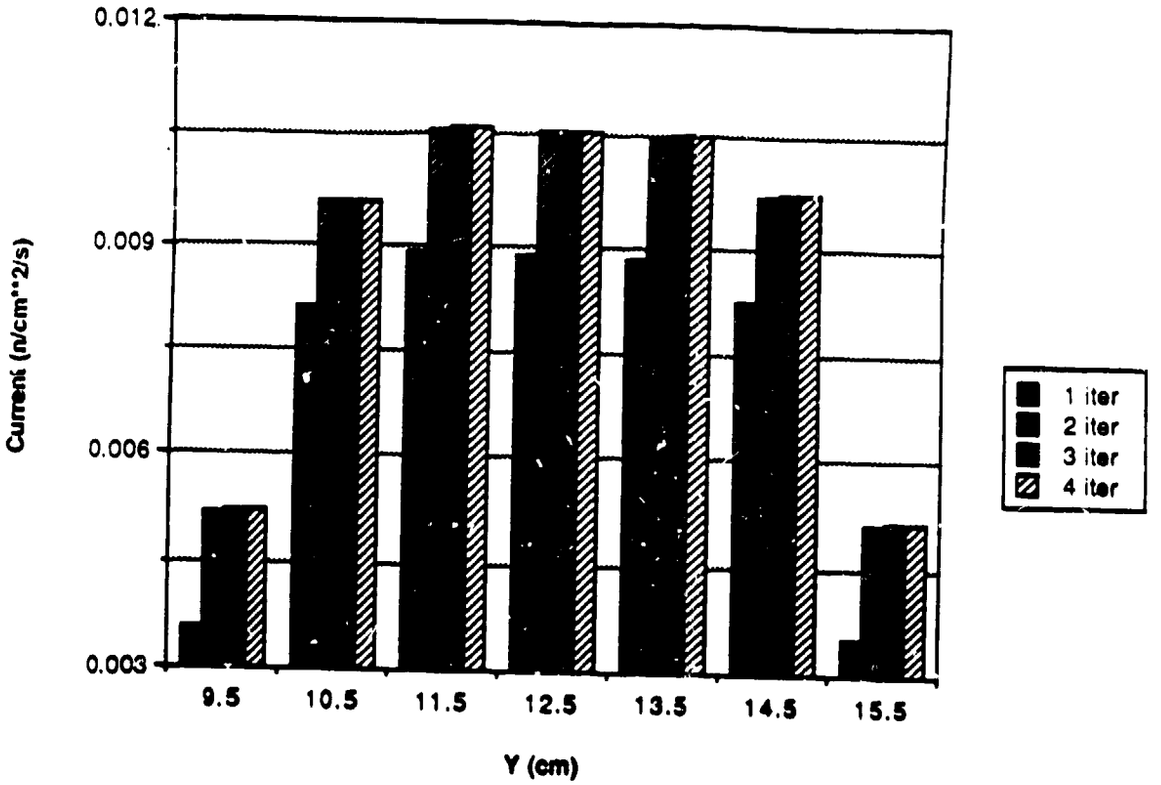

Fig. VII. Leakage current as a function of iteration from example 1.

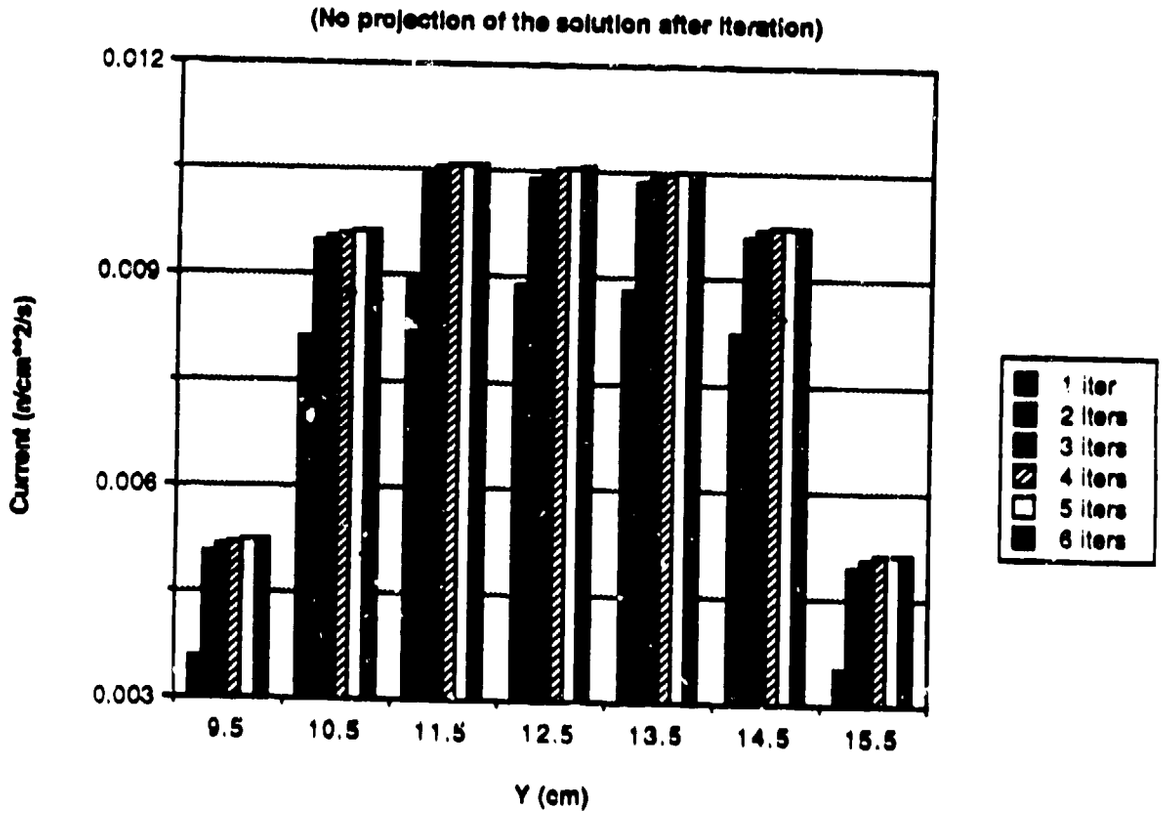

Fig. VIII. Leakage current as a function of iteration from example 1.

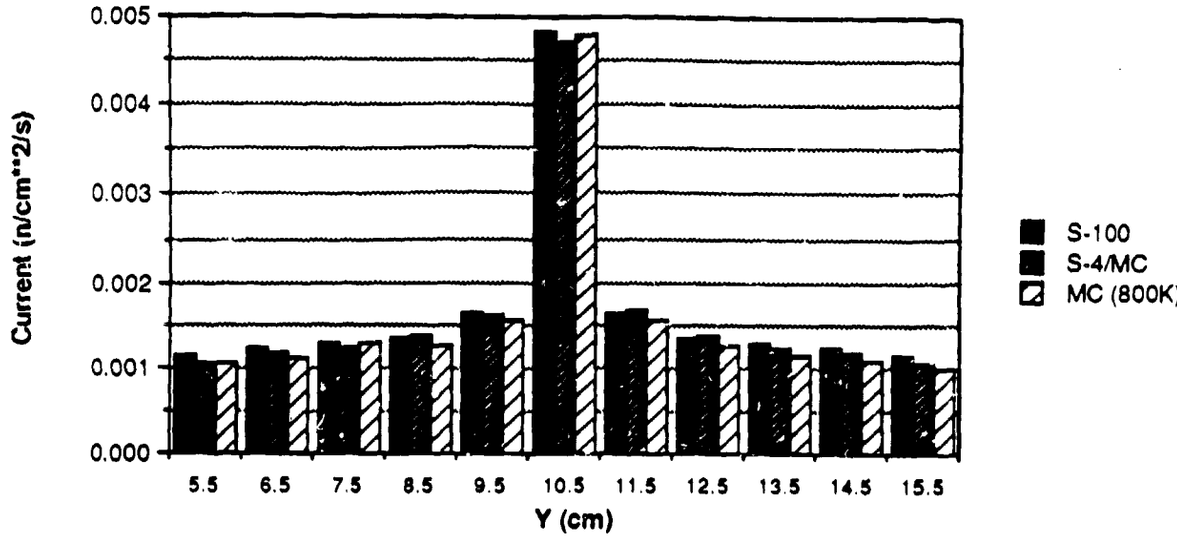

Fig. IX. Leakage Current from Example Problem 2.

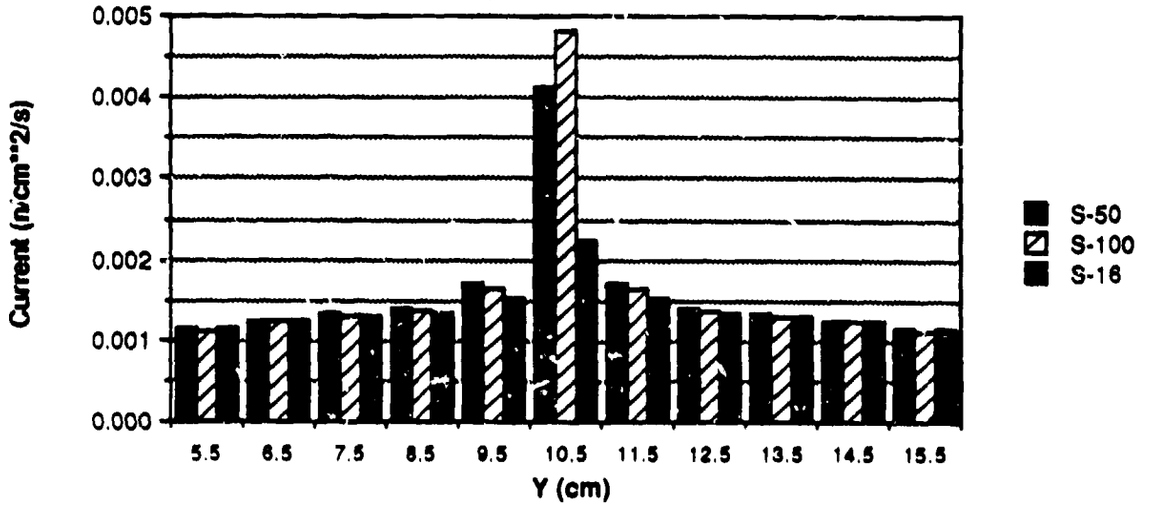

Fig. X. Leakage Current for Various Sn Ordrs from Example 2.

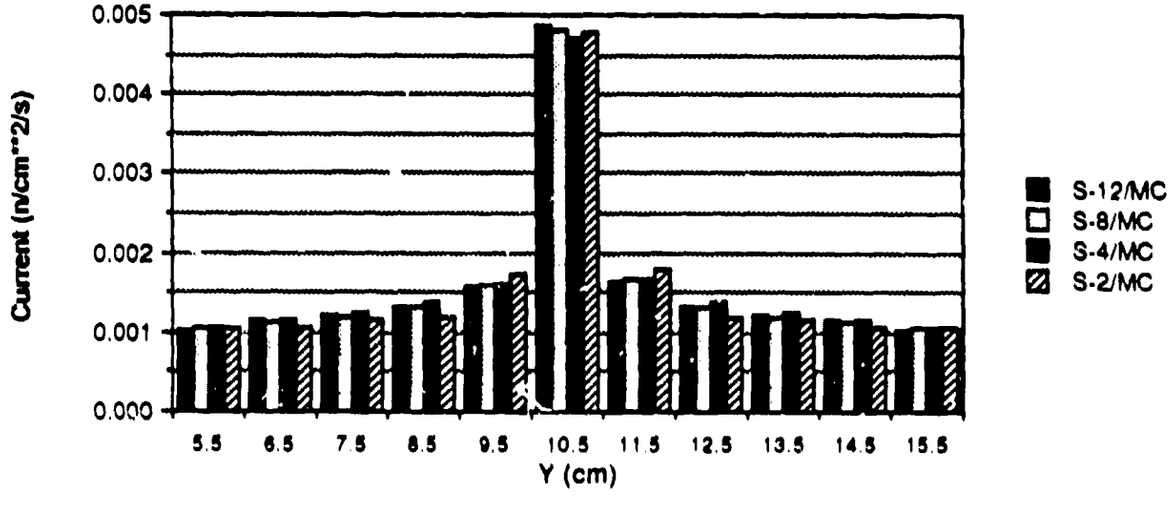

Fig. XI. Leakage Current as a Function of Hybrid Sn Order for Example 2.

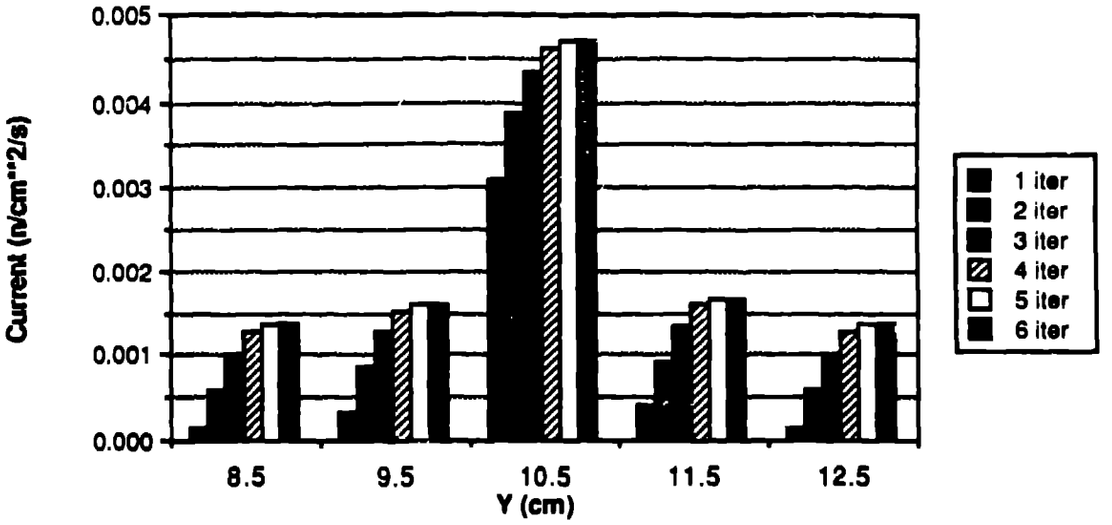

Fig. XII. Leakage Current as a Function of Iteration from Example 2.

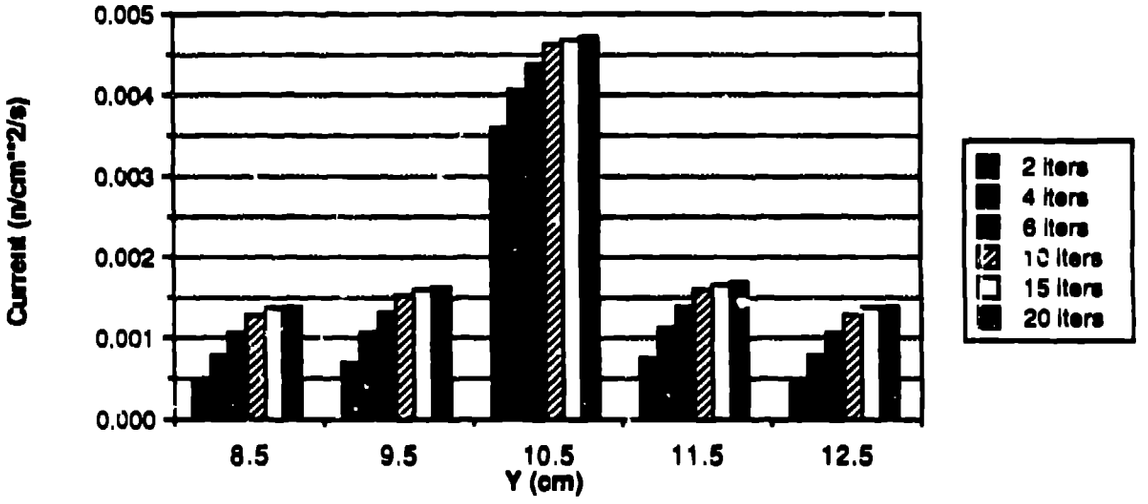

Fig. XIII. Leakage Current as a Function of Iteration from Example 2. (No projection of solution after iteration)

To show the advantage of using a boundary layer region in the Monte Carlo calculation, we present the solution as a function of S_N order in Fig. XIV. There, the data labeled S_N/MCO are from a calculation without a boundary region. We note the sensitivity of the hybrid solution to S_N order when no boundary layer is used. Even at S-12, the error at the slot is -20% compared with the benchmark and the hybrid method with the boundary layer.

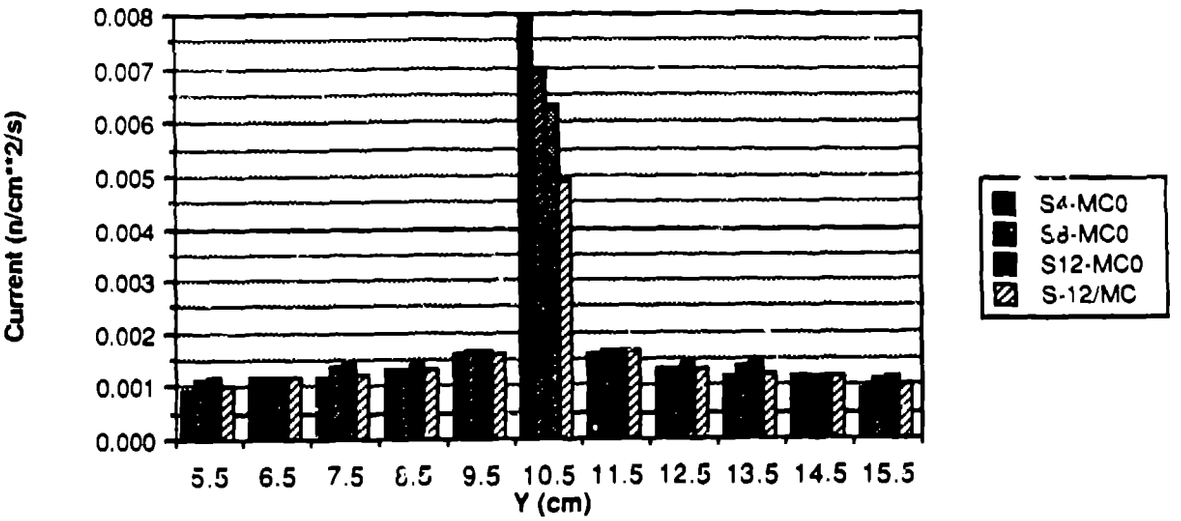

Fig. XIV. Leakage Current from Example 2 without a Boundary Layer region.

VI. OBSERVATIONS AND CONCLUSIONS

From the results of the above examples, we make the following observations:

1. An accurate answer for relatively simple problems is obtainable by an iterative linking of Monte Carlo and S_N through boundary response matrices.

2. The use of a boundary layer region is essential in obtaining results that are relatively insensitive to the S_N order used for the S_N regions.

3. The implementation of a Monte Carlo region into an S_N code need not interrupt the normal mesh-angle sweeping algorithm. Vectorization for this algorithm is not inhibited, and diffusion synthetic acceleration of the transport iterations is as effective as in the normal case.

4. The S_N/Monte Carlo boundary iteration procedure converges well and is effectively aided by the projection method described in Section II.D.

These observations lead us to conclude that the S_N/Monte Carlo hybrid method is very attractive for solving problems that neither method, by itself, can do very efficiently. It brings together the power of both methods in a systematic and convenient manner. In our implementation of this method, we have restricted it to compute geometrically very simple systems. However, the Monte Carlo algorithm is readily generalized, which will remove the geometric restriction in the Monte Carlo region. Of course, the S_N regions still must be described by either X-Y, R-Z, or R-θ geometries since we are in the TWODANT code. This code also restricts us to the multigroup approximation in energy, which reduces somewhat the power of the Monte Carlo method. However, the main concern is the storage required for the response matrices that go as the square of the number of spatial mesh points times angular mesh points on the interface between the two regions. For example, if we assume we have a maximum of 1.5×10^6 words of memory for response matrix storage,

then for a general problem with an asymmetric Monte Carlo region, we obtain the following permissible boundary spatial mesh as a function of S_N order

a) S_2, IL = 512,

b) S_4, IL = 204,

c) S_6, IL = 102,

d) S_8, IL = 61,

where IL is the number of spatial mesh intervals on the boundary. Thus, at S_8, we are beginning to reach our limit for reasonably sized, realistic problems.

REFERENCES

1. W. L. Filippone, R. E. Alcouffe, "Linking Monte Carlo and S_N via Response Matrices," Los Alamos National Laboratory report LA-10690-PR (Sept. 1985).

2. R. E. Alcouffe, et al., "User's Guide for TWODANT: A Code Package for Two-Dimensional, Diffusion Accelerated, Neutral Particle Transport," Los Alamos National Laboratory report LA-10049-M, Rev. 1 (1984).

3. R. E. Alcouffe, "Diffusion Synthetic Acceleration Methods for the Diamond Differenced Discrete Ordinates Equations," Nucl. Sci. Eng. 64, (1977).

4. W. L. Filippone, Trans. Am. Nucl. Soc. 40, 649 (1980).

5. Los Alamos National Laboratory Group X-6, "MCNP--A General Monte Carlo Code for Neutron and Photon Transport," Los Alamos National Laboratory report LA-7396-M, (1979).